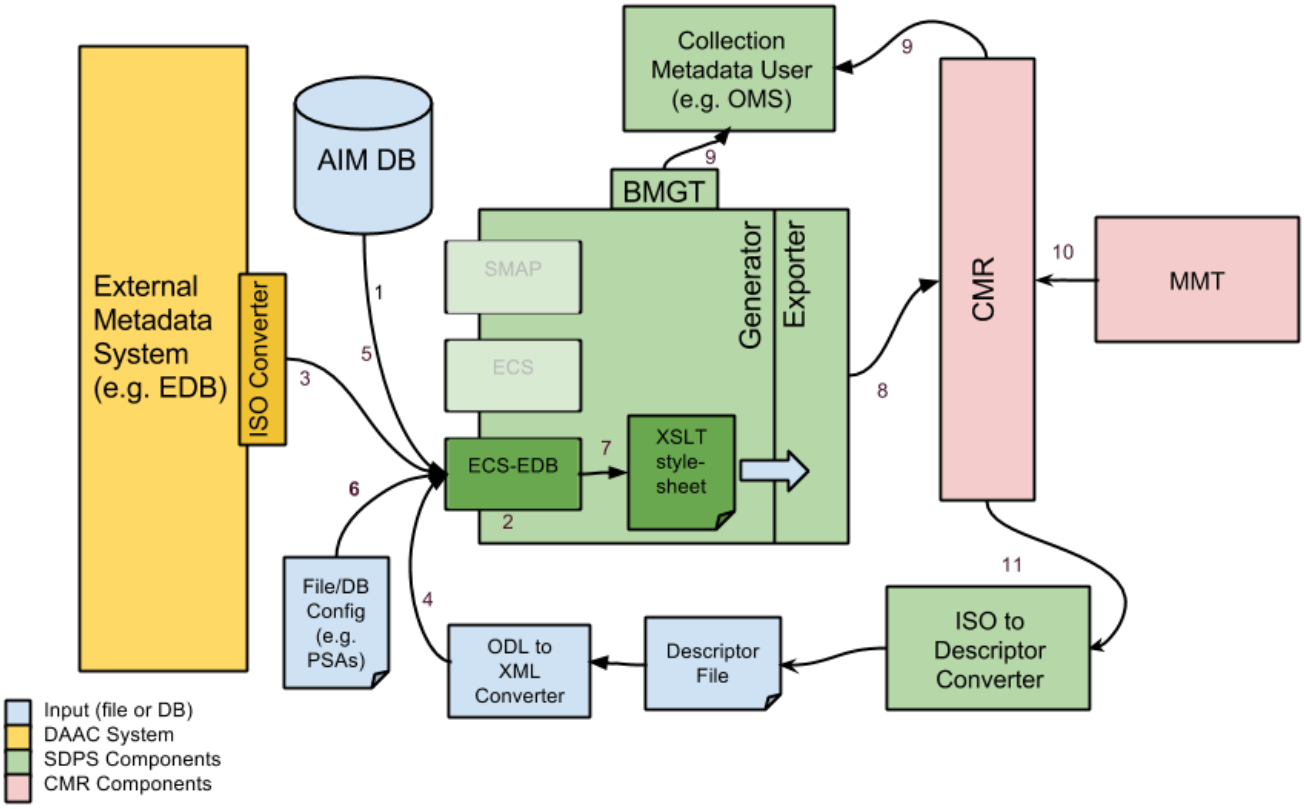
Proposal for BMGT to retrieve metadata from EDB

New Workflow for exporting collection metadata



1. BMGT pulls record from collection database table in response to an export request.
2. Based on the DataModelType column, BMGT uses the ECS-EDB generator implementation (as opposed to 'ECS' or 'ISO' (SMAP)).
3. BMGT Generator sends a request to the EDB ISO converter endpoint and receives the base metadata for the collection.
    a. Alternatively, BMGT could pull and existing metadata record from CMR and then export a new revision, with any additional metadata pulled from other sources.
4. BMGT Generator obtains the ECS descriptor file, converts it from ODL to XML, and retrieves any fields not contained in base metadata.
5. BMGT Generator queries the AIM database to obtain any fields configured there which are not in the base metadata or need to be modified (e.g. collection end date).
6. BMGT Generator reads any necessary additional configuration files to populate any remaining fields (e.g. PSAs).
7. BMGT passes the additional fields obtained above to an XSLT stylesheet which inserts them into the ISO data model.
    a. BMGT will need to resolve any conflicts between the various inputs before invoking the stylesheet.
8. BMGT Exporter exports the metadata to CMR.
9. Any entities which require the collection metadata can obtain it either by querying BMGT or CMR.
10. The DAAC makes updates to the metadata in CMR using the MMT (Metadata Management Tool) or some other means.
11. The ECS ISO to Descriptor Converter is run to retrieve the updated metadata in Descriptor ODL format and update the existing ECS descriptor
    a. Alternatively, two consecutive versions of the metadata could be pulled from CMR and diff-ed to create a patch file to apply to the existing descriptor - allowing items which exist only in the descriptor and no where else to remain untouched.
    b. This could be based on a subscription on the CMR so that any changes made via MMT would be propagated to ECS automatically.

Additional Information

- Items added and modified by BMGT currently for ECS collection metadata
- BMGT Configuration and Architecture

Assumptions:

- EDB implements an ISO(or ECHO10, ECS, UMM-C) endpoint.
- ISO is the desired format. If not, the above can also be performed with ECS/ECHO10 or UMM-C metadata.
- The descriptor schema is updated to support longer abstract field and other current incompatibilities with EDB.
- The EDB metadata is the 'base' version which other inputs should be applied to. If not, the ECS metadata could be the base and relevant fields would be pulled from EDB metadata and inserted into it.

Notes:

- This process can be generalized to the other DAACs with minimal modification. BMGT would simply need to implement a new collection metadata class in the generator to handle the idiosyncracies of different data providers, but assuming they are the same format (ISO or ECS), there will likely be a large amount of overlap and hence shared code.
- BMGT will need to resolve any conflicts between the same field in all the inputs. This sounds like it could be problematic, but in reality will probably be pretty simple. Consider the following assumptions which if true will greatly reduce the need for complex reconciliation between sources
    - The EDB record is correct - There should not be any fields that must be overridden elsewhere, as NSIDC will correct any problematic values in the EDB rather than overriding them in the descriptor, DB or other input file. One possible exception to this is the collection insert/update time, which we may want to get from the AIM DB, but this is simple (and done for ECS/SMAP metadata today)
    - BMGT is only inserting fields which are not in the EDB schema or unknown to EDB - BMGT will be inserting URLs, TwoDCoordinateSystem, etc. These values will likely not be in EDB, and BMGT already has the mechanisms to insert these in both ECS and SMAP metadata.
    - Each additional field will only be defined in one place - BMGT will look in only one place for each field (descriptor, DB, or config file), so there will not need to be any logic to determine which of multiple sources to use for a particular field (this implies that if the field DOES exist elsewhere, it will be ignored - there will be one definitive source per field.).
    - The Descriptor is only referenced for items not in the EDB descriptor - Anything that can be updated in the EDB should be. If the descriptor gets out of sync with EDB, that is fine, so long as the few fields (if any) which BMGT is extracting from it are up to date.
- This proposal makes the descriptor potentially obsolete for the purposes of collection metadata transfer. It is however still useful for configuring the collection in the ECS system, setting up validation rules, etc. If it is desired to keep the descriptor up to date, updates can be applied back to it from CMR as described above.
- Any changes made to the CMR collection metadata via MMT or some other means would need to be reflected in the EDB to avoid it being overwritten. The EDB update is outside of our scope (though it would be conceivable to update BMGT to be capable of exporting to EDB in addition to ECHO/CMR).

New Work

The following work will be required to implement the above:

- Implementation of an ISO (or ECS) endpoint in the EDB which BMGT can access to retrieve base metadata.
- Addition of a new DataModelType value in the AIM Collection table and the update of ESDTs to contain this new value.
- Implementation of a BMGT Collection class for the new DataModelType (would be an extension of the existing ECS or SMAP base classes)
- Creation of any config files/DB tables necessary for the storage of additional fields (e.g. PSA definitions and mappings)
- Implementation of an ISO to descriptor convertor.
- Optional - Implementation of an HTTP endpoint for a client to easily request metadata for a specified collection from BMGT(currently this exists only as a command line). This would be used by any SDPS components which currently use the descriptor to get collection metadata.